(12) UK Patent Application (19) GB (11) 2 336 276 (13) A

(43) Date of A Publication 13.10.1999

(21) Application No 9802124.9

(22) Date of Filing 30.01.1998

(71) Applicant(s)
SGS-Thomson Microelectronics Limited
(Incorporated in the United Kingdom)
1000 Aztec West, Almondsbury, BRISTOL, BS32 4SQ, United Kingdom

(72) Inventor(s)
Alan Lloyd

(74) Agent and/or Address for Service
Page White & Farrer
54 Doughty Street, LONDON, WC1N 2LS, United Kingdom

(51) INT CL[6]
H04L 29/06, H04N 5/00 7/20

(52) UK CL (Edition Q)
H4P PF PPEC
H4R RCT

(56) Documents Cited

GB 2309619 A	GB 2308281 A	EP 0821507 A2
EP 0821505 A1	EP 0742662 A1	EP 0727909 A2
EP 0680186 A1	WO 97/25665 A1	US 5619528 A
US 5533018 A	US 5530434 A	US 5420916 A

(58) Field of Search
UK CL (Edition Q) H4P PF PPEC PT
INT CL[6] H04L 25/02 29/06, H04N 5/00 7/20
Online : EPODOC

(54) Abstract Title

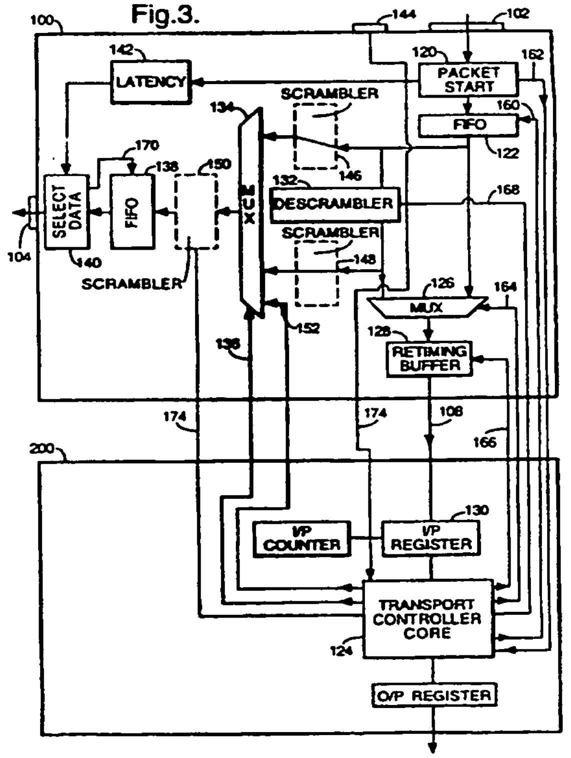
Protocol conversion

(57) A device 100, e.g. for use in a set-top box, receives a stream of data having a first protocol (via input 102). A portion of data to be output in a second, different protocol is selected. Processing means 200 processes selected data so that the selected data has a second protocol and the selected data having the second protocol is output by output means 104.

Both protocols can conform to the same standard, MPEG-2. Alternately there may be different standards, e.g. MPEG-2 and ATM format.

The data stream includes a plurality of different TV programs. Scrambling/descrambling is discussed. Stuffing bits are added to incomplete packets to fill them when reformatting.

Fig.3.



At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

GB 2 336 276 A

Fig.1.

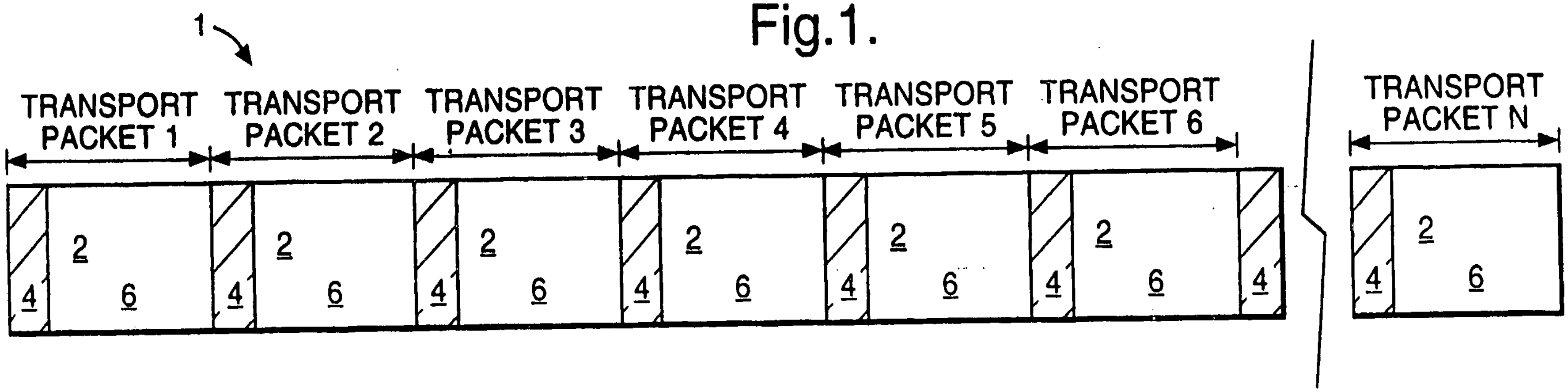

1
TRANSPORT PACKET 1
TRANSPORT PACKET 2
TRANSPORT PACKET 3
TRANSPORT PACKET 4
TRANSPORT PACKET 5
TRANSPORT PACKET 6
TRANSPORT PACKET N
2
4
6

Fig.2.

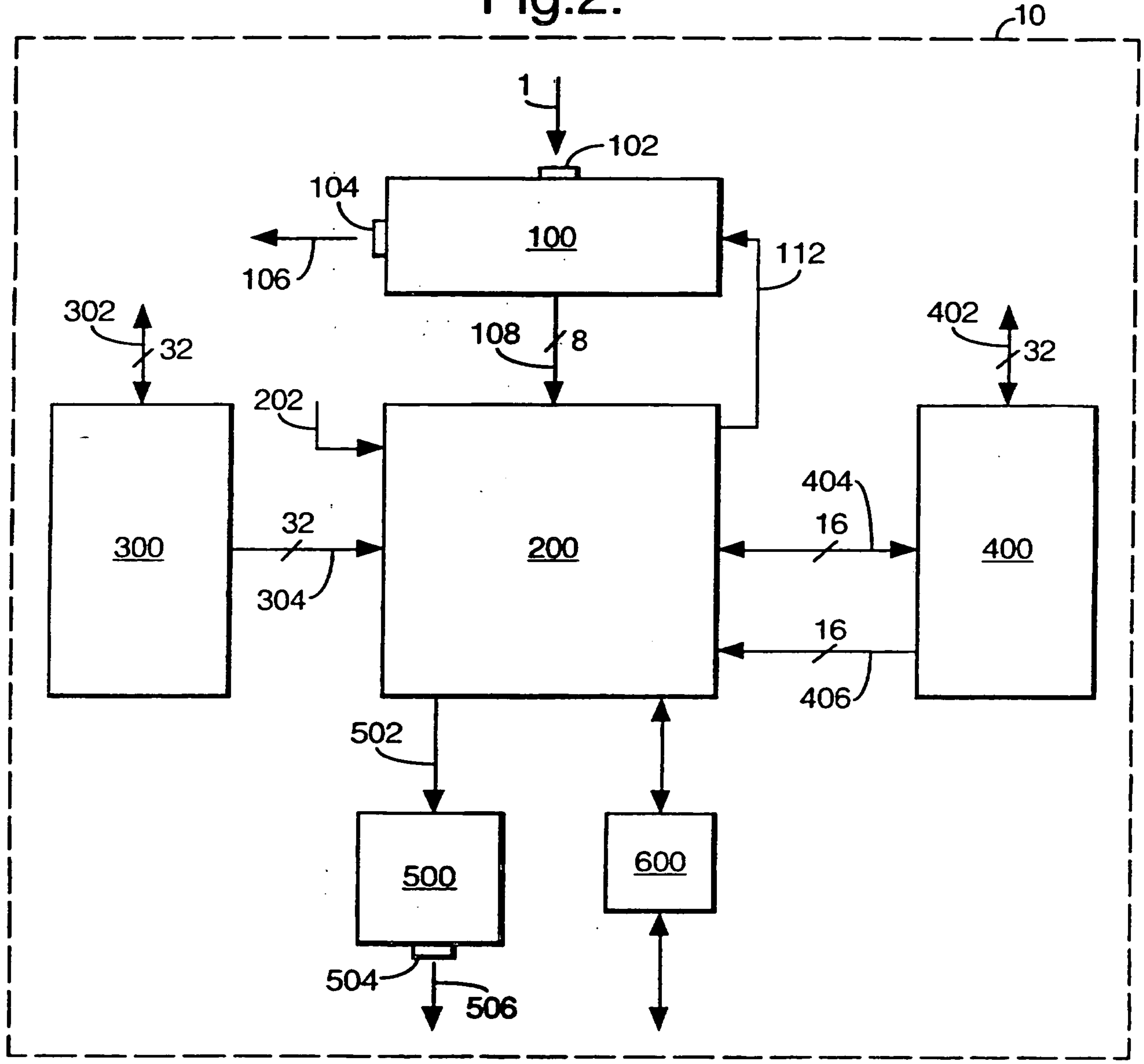

10
1
102
104
100
106
112
302
32
402
32
108
8
202
300
32
304
200
404
16
400
16
406
502
500
600
504
506

Fig.3.

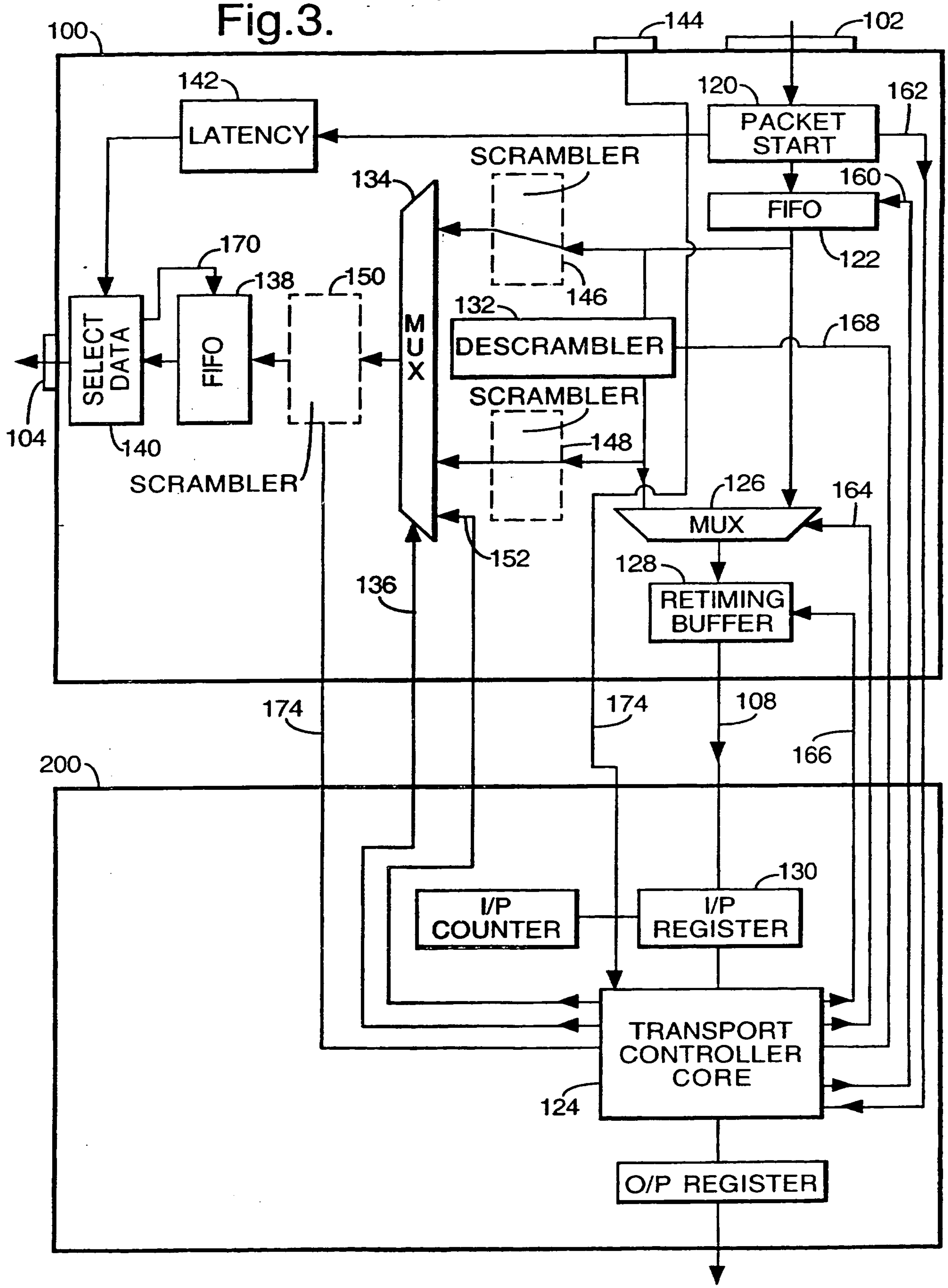
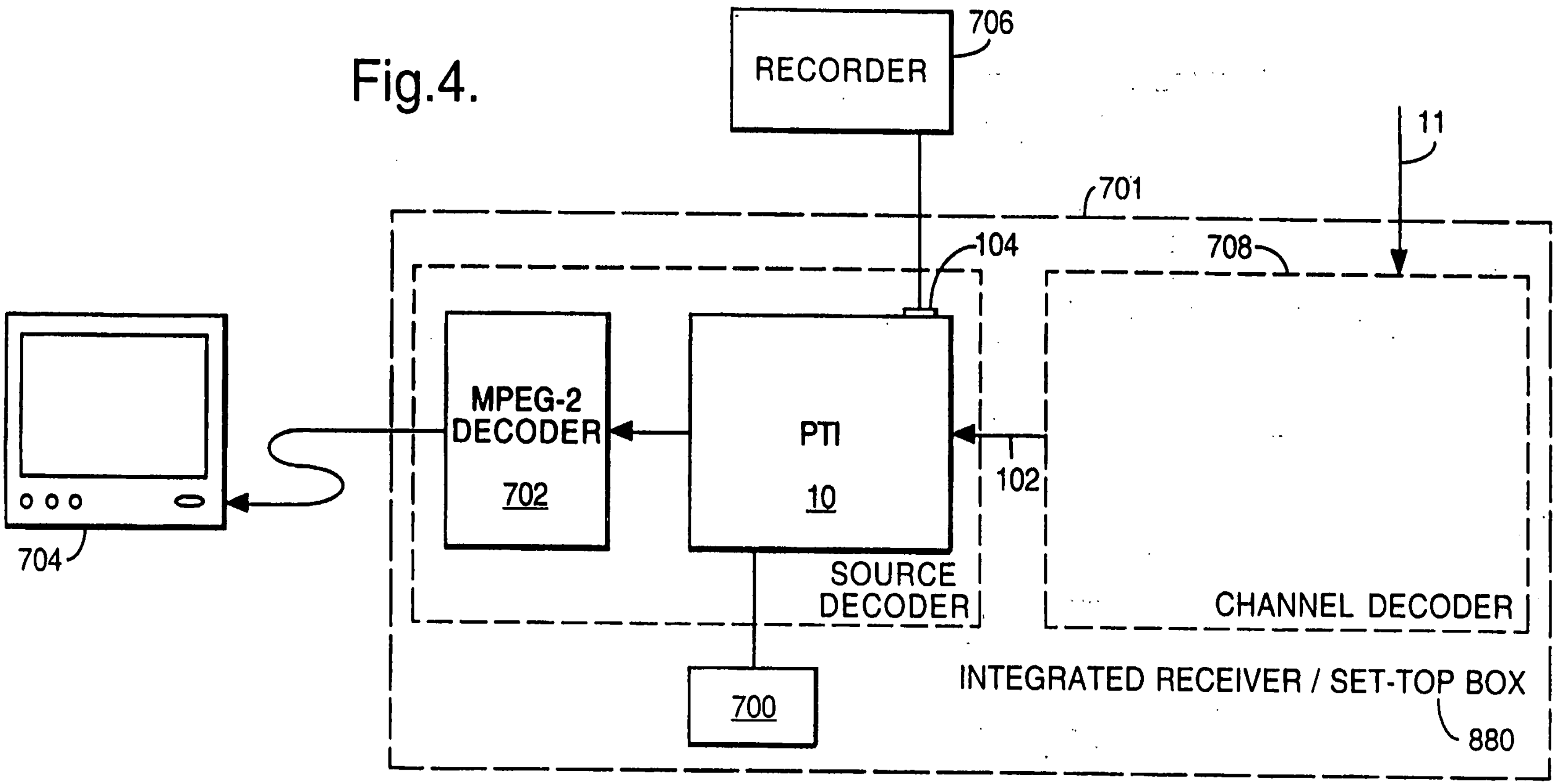

100
142
LATENCY
144
102
120
PACKET START
162
SCRAMBLER
134
160
FIFO
122
170
138
150
146
132
DESCRAMBLER
168
SELECT DATA
FIFO
MUX
SCRAMBLER
148
104
140
SCRAMBLER
126
164
MUX
152
128
RETIMING BUFFER
136
174
174
108
166
200
130
I/P COUNTER
I/P REGISTER
TRANSPORT CONTROLLER CORE
124
O/P REGISTER

Fig.4.


RECORDER
706
11
701
104
708
MPEG-2
DECODER
702
PTI
10
102
SOURCE
DECODER
CHANNEL DECODER
INTEGRATED RECEIVER / SET-TOP BOX
880
700
704

DEVICE AND METHOD FOR PROTOCOL CONVERSION

The present invention relates to a device and method for converting at least a portion of a stream of data having one protocol into one having a second, different protocol.

Set top boxes are used, for example in the context of cable television and satellite television. A set top box is arranged to receive television programmes from a satellite or via a cable and to output a programme which is displayed on a television screen or recorded on a video recorder. With both cable and satellite television, an input stream is received at an interface of the set top box. The input stream is generally scrambled and comprises audio and visual information about several different television programmes, the information being time multiplexed together. Control information will also be included in the received input stream. Information relating to a television programme selected by the user is demultiplexed by the set top box from the input stream to provide the selected programme which is then output by the set top box to for example a television screen, video recorder or indeed any other type of recorder.

Cable and satellite televisions generally include pay television services where a user will pay for a selected number of channels, a particular event or one off program such as a football match, concert, sporting event, or the like, or a particular film or the like.

The information for the programmes which are on channels for which the user has not paid remain scrambled and cannot be unscrambled. For those channels with a particular programme for which the user has paid, the control information will include information to allow descrambling of the received signal. Typically, the descrambling operation will also require descrambling information included on a smart card or the like. The smart card is replaced at regular intervals for security.

However, with certain services, the smart card is not used.

These pay television services often include programmes which are unsuitable for viewing by, for example, children. Furthermore, it is often the case that these types of programmes will be transmitted late at night and the user may therefore wish to record the programme for subsequent viewing. The problem then exists that children may inadvertently view this unsuitable material which has been recorded on a video or the like. A further problem is that the provider of a pay television service may wish to avoid the problem of a user recording a programme and that programme subsequently being duplicated for commercial purposes.

One problem which has been recognised by the applicant is that a number of different standards and different protocols have been proposed for the data stream which is received by a set top box. Indeed, some users of the same standard have modified the protocol for that standard to suit their own requirements. Furthermore with digital television still in its infancy, changes are continuously being made to the protocols which are used to optimise them. However, recorders or the like which are connected to the set top box may only be able to deal with one protocol which may be different to that of the received data stream. This may arise if a user decides to change the satellite or cable service provider which he uses. If the new service provider uses a different standard or protocol, the user would have to buy a new recorder which can deal with the new protocol. This problem may arise if the user's existing service provider changes the protocol or standard which they use.

Accordingly, it is an aim of certain embodiments of the present invention that this latter problem be addressed.

According to one aspect of the present invention, there is provided a device for receiving a stream of data having a first protocol, said device comprising means for selecting at least a

portion of data to be output in a second, different protocol; processing means for processing the selected data so that said selected data has said second protocol; and means for outputting the selected data having the second protocol.

In this way, if a device which is connected to the output means can only deal with data having a different protocol to that of the stream of data of the first protocol, that further device can be connected to the receiving device.

The stream of data preferably comprises a stream of packets of data. Thus, it is possible for the receiving device to change the protocol of the received packets of data to produce packets of data having a different protocol. Thus, the protocol of individual packets can be changed.

The length of the data packets of the first protocol may be shorter than the length of the data packets of the second protocol and the processing means may be arranged to include data from a packet of said selected data having said first protocol and stuffing bits in a packet of data having said second protocol, said outputting means being arranged to output said packet of data in the second protocol. In this way, packets of data in the first protocol can be simply modified in order to provide packets of data in the second protocol.

The length of the data packets of the first protocol may, alternatively, be longer than the length of the data packets of the second protocol and the processing means may be arranged to include data from a packet of said selected data having the first protocol in a plurality of packets of data having said second protocol, said outputting means being arranged to output said packets of data in the second protocol. The number of packets of data required in the second protocol will depend on the respective length of the packets in the first protocol and the second protocol. Preferably, when one of the plurality packets in the second protocol is not full with data from the packet of

selected data in the first protocol, said processing means is arranged to include stuffing bits in said packet which is not full to thereby fill that packet. Alternatively, when one of the plurality of packets in said second protocol is not full with data from said packet of selected data in the first protocol, said processing means is arranged to include data from the next packet of said selected data to thereby fill said one packet.

Preferably, the data stream includes a plurality of different programmes. These programmes may be television programmes or the like. In a preferred embodiment of the present invention, only one of the plurality of programmes is output by said outputting means. A selected programme may be output on a different output of said device. This selected programme may be the same or different to the selected data output by said outputting means. The output provided on the different output will conform to the first protocol although the received data packets may have been processed.

The programme output from said output means may be output at the same time as the programme output from said different output. In this way, a user is able to view one programme whilst recording a different programme.

Preferably the packets of the first protocol and the second protocol each have a header part and a payload part and when converting the packets of the first protocol to packets of the second protocol, the header part of the packet having the first protocol is modified to conform to the second protocol. The payload part may not require modification in order to convert from the first protocol to the second protocol.

Detecting means may be provided for detecting the beginning of each packet having said first protocol and the timing of said device is dependent on the detection of the beginning of each packet. The detecting means may be arranged to detect at least a portion of the header to detect the beginning of the packet.

In a preferred embodiment of the present invention, the selected data is processed in real time. Preferably, one packet of data in the first protocol is converted to one packet of data in the second protocol before the next packet of data is received by the receiving device.

Preferably, the packet of data in the second protocol is output when the beginning of the next packet having the first protocol is received. In this way, the timing of the device may be controlled by the detection of the beginning of each packet.

Preferably, the device comprises an input interface having an input for receiving said stream of data and said outputting means, and a controller having said processing means.

Preferably, the input interface comprises means for descrambling the received stream of data. Thus, the received stream of data may be descrambled before being modified to conform to the second protocol. The input interface is preferably controlled by said controller. The controller may control the descrambling means, the detecting means and the output means. The detecting means may be arranged to provide an output signal to the processor to control the timing of the device. Preferably, the signal provided by the detecting means is used to control the output via said outputting means. The outputting means may include buffer means. This may allow data to accumulate in the buffer means until a complete packet is ready to be output from the output means or while the processor is performing the protocol conversion.

The processing means comprises a programmable processor. In this way, it is simple to alter the operation of the processing means so that it can deal with different protocols.

The data stream preferably comprises a digital data stream. The first and second protocols may both conform to the MPEG-2 (Moving Picture Expert Group) standard coding protocol or similar protocol.

Preferably, the device forms part of a set top box.

According to a second aspect of the present invention, there is provided a device for use in a set top box comprising means for receiving a stream of packets of data having a first protocol and including information relating to a plurality of different programmes; means for processing the received stream of data to identify selected packets of data relating to a programme to be output, said processing means being arranged to process the selected data packets so the selected packets conform to a second, different protocol; output means for outputting the selected data packets in the second protocol.

For a better understanding of the present invention and as to how the same may be carried into effect, reference will now be made by way of example to the accompanying drawings in which:

Figure 1 schematically shows a transport stream;
Figure 2 shows a schematic diagram of a programmable transport interface embodying the present invention;
Figure 3 shows a block diagram of part of the input interface and part of the transport controller of the programable transport interface shown in Figure 2; and
Figure 4 shows a set top box incorporating the programmable transport interface of Figure 2 and which is connected to a recorder and a screen.

Figure 1 illustrates a portion of the transport stream 1 (data stream) which is composed of a series of N transport packets 2. Each transport packet 2 comprises a transport packet header 4 and a transport packet payload 6. The transport stream is a bit stream which carries in the transport packet payloads 6 of information for recreating, for example, a number of different television programmes. The transport stream is formed by source encoding the television programmes. The transport stream is then typically channel encoded for transmission for example by satellite or cable and channel decoded at a respective receiver

to reproduce the transport stream. The transport stream is then source decoded to recreate a selected one of the different television programmes transmitted by the transport stream 1. Each particular television programme may require four different types of information in order to recreate the programme. That information may consist of audio information, video information, descrambling information and tables of programme information. Each transport packet 2 is preferably but not necessarily associated with one particular television programme, one particular source encoding time and one of the four different types of information. The individual transport packets are time division multiplexed to form the transport stream and allow the real-time recreation of any one of the different television programmes from the transport stream. To recreate a television programme, the transport stream 1 is demultiplexed to recover only the transport payloads 6 of audio information, video information, descrambling information and tables of programme information which are associated with the selected television programme. The recovered payloads are then decoded to recreate the television programme. In general, only the payloads will be scrambled and not the headers.

According to one digital broadcast standard DVB (Digital Video Broadcasting) each of the transport packets is 188 bytes long of which the transport packet header is 4 bytes long. The payload 6 contains packetised information, such as the information for recreating a number of different television programmes as discussed hereinbefore. With this known standard, the audio and video information in the payloads 6 have been packetised and encoded in accordance with the MPEG-2 compression standard. A programable transport interface 10 (PTI), which is illustrated in Figure 2, is used to process the received transport stream 1 and produce a data output stream 506 suitable for reconstitution as a television programme after MPEG-2 decoding by MPEG-2 decoders 702 (see Figure 4). The programmable transport interface 10 is included in a receiver or set top box 701 which receives the transport stream 1.

The transport packet header 4 contains a synchronisation byte which identifies the beginning of each transport packet 2. The transport packet header 4 also contains a packet identification PID which identifies the information type and the television programme associated with the transport packet payload 6. The transport packet 2 also contains information identifying the source encoding type of the transport packet. The transport packet header 4 including the synchronisation byte and the PID is not scrambled. The transport packet payload 6 may itself be scrambled.

The programmable transport interface 10 shown in Figure 2 also produces an alternative output stream 106 which will be described in more detail hereinafter. This alternative output stream 106 may be an output derived from the transport stream 1 which has been modified for example by encryption or by changing the communication standard or protocol under which the transport stream has been prepared.

The programmable transport interface PTI 10 performs the following functions amongst others. The PTI 10 uses the synchronisation byte to identify the start of a transport packet 2 and uses the packet identification PID to identify the type of information contained in the packet and the television programme it represents. The PTI 10 descrambles, if necessary, the transport packet payloads 6 and demultiplexes the transport stream 1 to produce the data output stream 506, this data output stream comprising a stream of audio information associated with a selected television programme, a stream of video information associated with the selected television programme and tables of programme information associated with the selected television programme. The PTI 10 then outputs these streams to the necessary decoders 702 and/or to buffers in an external memory (not shown) to reproduce the selected television programme.

The PTI 10 comprises six functional blocks: the input interface 100; the transport controller 200; the instruction SRAM (static

random access memory) 300; the data SRAM 400; the multi-channel DMA (direct memory access) 500; and the controller and status register interface 600.

The input interface 100 has a transport stream input interface 102 for receiving the transport stream 1 and an alternative stream output interface 104 for outputting the alternative output stream 106. The function of the input interface 100 will be described in more detail hereinafter.

The transport controller 200 receives from the input interface 100 via interconnect 108 the transport packet header 4 of the transport packet arriving at the transport stream input interface 102. The transport controller 200 uses the packet identification PID in the transport packet header 4 to determine whether the transport packet 2 entering the input interface 100 via the transport stream input interface 102 is associated with a selected television programme. If it is not, the received transport packet 2 is discarded. If it is, the transport controller 200 controls the input interface 100 to descramble and supply the transport packet payload 6 via the interconnect 108 to the transport controller 200. The transport controller 200 may pass a payload 6 associated with audio or video information for the selected programme straight to the multi-channel DMA 500 via interconnect 502. If the payload 6 relates to a section of a table of programme information, the transport controller 200 may process the information before providing that information at its output 502. Alternatively, the packet 6 may be output, after processing by the transport controller 200, via the alternative stream output interface 104. This will be described in more detail hereinafter.

The transport controller 200 comprises a processor in the form of a transport controller core 124 (see figure 3) which reads instructions from the instruction SRAM 300. The transport controller 200 is connected to the SRAM 300 by interconnect 304 and reads instructions from the SRAM 300 via the interconnect

304. A system processor 700 (see Figure 4) may read and write to the instruction SRAM 300 via the interface 302 allowing the transport controller instructions to be varied.

The data SRAM 400 can be read from and written to by the transport controller core 124 of the transport controller 200 via the interconnect 404. A search engine (not shown) within the transport controller 200 reads from the data SRAM 400 via interconnect 406. The search engine associates a pointer with each of the programme identifications PIDs in the transport packet headers 4. The data SRAM 400 stores at a location indicated by the pointer information associated with a transport packet 2 having a particular PID. This information is read over interconnect 406 and it enables the transport controller to control the production of input interface control signals 112 and the processing of the bits received on interconnect 108. The data SRAM 400 can be written to and read from the system processor 700 via the interface 402. The transport controller 200 produces a transport controller output which is supplied to the multi-channel DMA 500 via interconnect 502. The multi-channel DMA 500 has an external memory interface 504 which supplies the data output stream 506 to decoders 702 or an external memory.

Reference will now be made to Figure 3 which shows part of the input interface 100 and part of the transport controller 200 in more detail. The input interface 100 is arranged to receive the transport stream 1 via transport stream interface 102. The transport stream received via transport stream interface 102 is passed to packet start block 120 which is arranged to look for the synchronisation byte of each transport packet 2 which identifies the beginning of each packet. In the start up mode, the packet start block 120 looks at the input stream until it finds a synchronisation byte. In order to establish that what is located is the synchronisation byte and not, for example, part of the payload 6 which happens to contain a sequence of bits identical to that of the synchronisation byte, the packet start block 120 checks to see that a synchronisation byte is present

a predetermined number of bytes later, i.e. at a location corresponding to the beginning of the next packet. In this embodiment, the packet start block 120 only checks for the occurrence of two synchronisation bytes spaced apart by a predetermined number of bytes (corresponding to the length of the packet). However, in embodiments of the present invention, the packet start block 120 can check that the synchronisation byte occurs a predetermined number of times, each occurrence of the synchronisation byte being separated by the number of bytes contained in each transport packet. For example, in the DVB standard, the packet start block 120 checks to see that a synchronisation byte occurs every 188 bytes in order to confirm that the beginning of the transport packets has been identified. Once the packet start block 120 has verified that the beginning of a transport packet has occurred, the packet start block 120 provides an output via interconnect 162 to the transport controller core 124 indicative that the beginning of the transport packet has been identified.

A FIFO (First-In-First-Out register) 122 is connected to the output of the packet start block 120 and whilst the packet start block 120 is in the set-up mode, the FIFO 122 is controlled by the transport controller core 124 via interconnect 160 to simply allow the input transport stream to flow through that FIFO 122. The output of the FIFO 122 is connected to a multiplexer 126 which receives a control signal from the transport controller core 124 via interconnect 164. That multiplexer 126, in the set-up mode, is arranged to pass the output of the FIFO 122 therethrough to a retiming buffer 128, which is controlled by the transport controller core 124 via interconnect 166. In the set-up mode of operation, the retiming buffer 128 simply outputs the transport stream received from the multiplexer 120 to the transport controller 200. In particular, the transport stream is passed via an input register 130 of the transport controller 200 to the transport controller core 124. Until the transport controller core 124 receives the packet start signal from the packet start block 120, the transport controller core 124 simply

discards the received transport stream.

When the transport controller core 124 receives the signal from the packet start block 120 indicating that the beginning of a packet has been located, the transport controller core 124 provides an output signal via interconnect 160 to the FIFO 122. This control signal is such that once the transport packet header 4 has passed through the FIFO 122, the FIFO 122 is prevented from passing any more of the received transport stream therethrough. Instead, the payload 6 starts to accumulate in the FIFO 122. The transport packet header 4 is passed through the multiplexer 126 and the retiming buffer 128 to the transport controller 200. In particular, the transport packet header 4 is passed via the input register 130 to the transport controller core 124 which is arranged to process this header. The packet header 4 may contain information which can be used to process, if necessary, the transport packet payload. Firstly, the synchronisation byte is used to control the timing of the programmable transport interface 10. The transport packet header 4 also contains information as to whether or not the transport packet payload 6 is scrambled or not. If the payload 6 is scrambled, then the packet header 4 contains information about which key to use for descrambling the payload 6. The packet header 4 also contains the packet identification PID which identifies the information type contained in the payload 6 and the television programme carried by the associated payload 6.

The transport controller core 124 checks the transport packet header 4 to determine if the payload 6 contains information on a selected television programme. This selected television programme may be the programme to be provided by interconnect 502 to the multichannel DMA 500 for viewing by the user, for example. The selected programme may additionally or alternatively be that which is to be output via the alternative stream output interface 104 of the input interface 100. In embodiments of the present invention, the alternative output stream 106 may contain the same programme or a different programme to that output via

interconnect 502. However, in embodiments of the present invention the format of the output provided by the alternative stream interface may be altered. This will be described in more detail hereinafter.

If the transport controller core 124 determines from the packet header 4 that the payload relates to a selected programme, the transport controller core 124 determines from the header whether or not the payload requires descrambling. If it is determined that the payload is scrambled, then the transport controller core 124 is arranged to provide an output via interconnect 168 to the descrambler 132 of the input interface 100 including at least part of the necessary descramble key. The descramble key may be obtained from one or more of the following: smart card (not shown), the data SRAM via the transporter controller core, and the packet header. Once the transport controller core 124 has completed the processing of the packet header 4 for a transport packet 1 which contains a payload 6 relating to a selected programme, the transport controller core 124 provides an output signal to the FIFO 122 allowing the accumulated payload 6 to be output therefrom. The FIFO 122 in fact outputs the data stream both to the descrambler 132 and the multiplexer 126 directly. If the payload 6 contains unscrambled data, then the descrambler 132 will not be enabled by the transport controller core 124 and the multiplexer 126 will be arranged to output the data directly received from the FIFO 122. Alternatively, if the transport controller core 124 has determined that the payload is scrambled, the descrambler 132 will be enabled. The descrambler 132 will descramble in accordance with the descramble key received from the transport controller core, at least partially, the received payload and output the descrambled payload to the multiplexer 126. In these circumstances, the multiplexer 126 is controlled by the controller core 124 via interconnect 164 to select the output from the descrambler 132 as its input.

The output of the multiplexer 126 is output to the retiming buffer 128 which is controlled by the transport controller core

124 via interconnect 166. The retiming buffer 128 is in fact another FIFO and is used to achieve smooth flow control for the system as a whole. The retiming buffer 128 may be controlled by the transport controller core 124 to store the data received from the multiplexer 126 until a predetermined number of bits or bytes has been stored in the retiming buffer 128. When the number of bits or bytes in the retiming buffer 128 has reached the predetermined level, then those bytes may be output to the transport controller 200. The function of the retiming buffer 128 is twofold. Firstly, the retiming buffer 128 stores the data until such time that the transport controller core 124 is able to receive that data. This means that the descrambler 128 can continue to descramble even if the transport controller is not ready to receive the next byte of data. Secondly, the retiming buffer 128 is arranged to accumulate the data until the number of bits or bytes has reached a predetermined level. In some embodiments, optimum efficiency of the device is achieved if a given minimum number of bits or bytes is dealt with by the transport controller core 124 at one time.

The payload in the retiming buffer 128 will be output to the input register 130 of the transport controller and then to the transport controller core 124.

If the transport controller core 124 determines that the data packet contains data relating to an unselected programme, then that packet will be discarded.

The transport controller core 124 controls the FIFO 122 so that once the header of the next packet has passed therethrough, the FIFO 122 starts to accumulate the payload 6 of the next packet.

In one embodiment of the present invention, the transport stream 1 is received in one digital video broadcast standard having a particular protocol but for example, the video recorder or any other suitable recording device to be connected to the programmable transport interface 10 is only able to process data

received in a different protocol. With embodiments of the present invention, it is possible to receive a transport stream 1 in one protocol and convert it into another, different protocol. This will now be described in more detail hereinafter. One of the protocols or standards may be the DVB standard mentioned previously. One of the protocols or standard may alternatively or additionally be the DSS (Digital Satellite) standard which has 130 bytes in each packet. Another protocol or standard is the DVD (Digital Versatile Disc which is also known as Digital Video Disc) standard which has 4096 bytes in each packet.

Generally, the alternative output stream 106 provided by the input interface 100 will generally only relate to one programme carried by the transport stream. However, it should be appreciated that in some embodiments of the present invention, more than one programme may be output from the alternative output interface 104. The transport controller core 124 is arranged to check, as previously described, each transport packet header in order to identify whether or not the given packet contains information relating to the or a selected programme to be output via the alternative output 104. When the transport controller core 124 identifies a transport packet which contains information relating to the or a selected programme, the transport controller core 124 is arranged to change the format of the received data packet in order to meet the requirements of the protocol to be used for the alternative output stream 106.

It should be appreciated that in some embodiments of the present invention, a separate processor may be provided in order to deal with this re-protocolling of the received data packets. In these embodiments of the present invention, a switch may be provided in either the transport controller or the input interface 100 so that the payload is passed directly to the separate processor, when necessary. The separate processor could be provided in the input interface 100 or the transport controller 200.

However, in the preferred embodiment, the transport controller

core 124 is responsible for reformatting the received transport stream 1 to the desired protocol. There are two main types of situation. With the first situation, the transport stream uses a first protocol where the number of bytes in each data packet is less than that of the data packet having the second protocol to be output via the alternative stream output interface 104. In this situation, the transport controller core 124 is arranged to reformat the received packets, which have the first protocol, including both the header 4 and the payload 6. This may involve changing the position of the bits or bytes within the packet to ensure that they are in the correct position in the new packet. This may also involve the reordering of certain of the bits or bytes. This will be dependent on the two protocols themselves. To ensure that the packet output via the alternative stream output interface 104 has the required number of bytes, the transport controller core 124 inserts stuffing bits into the packet having the number of bytes of the first protocol to ensure that it has the required number of bytes. These stuffing bits may consist of 1s, 0s or a predetermined sequence of 1s and 0s. A single packet having the first protocol may be included in one packet of the second protocol. Where the packets of the first protocol are much smaller than the packets of the second protocol, two or more packets of the first protocol may be included in a single packet of the second protocol.

When the transport controller core 124 has completed the re-protocolling of the received transport packet 1, that packet is output from the transport controller core to a second multiplexer 134 of the input interface 100. That second multiplexer 134 is controlled via interconnect 136 by the transport controller core 124. In addition to the input received from the transport controller core 124, the second multiplexer also receives an input from the FIFO 122 and an input from the descrambler 132. The transport controller core 124 controls the second multiplexer 134 via interconnect 136 to select one of the inputs to the multiplexer 134. Where reprotocolling is to take place, the reformatted packet input from the transport controller 124 is

selected and the reformatted packet is output via the multiplexer 134 to a further FIFO 138. In a modification to this arrangement, the header 4 only is reprotocolled by the transport controller core 124 and the reprotocolled header will be output via interconnect 156 and the payload will be received directly from FIFO 122 or via the descrambler 132. The payload may be sent at the same time to the second multiplexer 134 and the transport controller core. The transport controller core 124 can therefore monitor what payload is being sent to the alternative output without itself needing to pass that data to the second multiplexer 134. The function of the optional scramblers 146,148 and 150 shown in dotted lines will be described in more detail hereinafter.

The reformatted transport packet is stored in the FIFO 138 until it is ready to be output. The FIFO 138 is controlled by a select data block 140. The select data block 140 in conjunction with a latency block 142 is arranged to ensure that the device as a whole avoids any latency problems. In particular, the receipt of the successive synchronisation bytes controls the timing of the programmable transport interface 10 as a whole. As the output of the alternative output stream 106 generally only relates to one programme, there is a certain amount of flexibility in the latency. In particular, the only criteria which needs to be satisfied is that on average one transport packet is output via the alternative output interface 104 before the next transport packet which is to be output on the alternative output stream is ready to be output. In other words the rate of output of the reformatted packets should not generally be slower than the rate at which the packets relating to the selected programme or programmes are received. As mentioned previously, the transport stream 1 will generally include information in different transport packets relating to a number of different programmes and not just the selected programme or programmes which will be output as the alternative output stream. In one preferred embodiment, the reformatted transport packet is output when a subsequent synchronisation byte is detected by the packet start

block 120.

Each time the packet start block 120 identifies the synchronisation byte, an output is provided to the latency block 142 which in turn provides an output to select data block 140. When the select data block 140 has received an output from the latency block 142, the select data block 140 provides an output signal via interconnect 170 to the FIFO 138 thus causing the reformatted transport packet to be output via the alternative stream output interface 104. The output of the latency block 142 thus acts as a clock signal for the output of the reprotocolled data packet.

In a particularly preferred embodiment, the programmable transport interface 10 is able to reformat the transport packet so that when the synchronisation byte of the packet immediately following the packet to be reformatted is received, the reformatted packet is output. This provides optimum latency to the device. However in other embodiments of the invention, there may be a delay equivalent to several cycles (i.e. several synchronisation bytes) before the reformatted packet is output. This is generally only possible in those embodiments where not all of the data stream is to be output via the alternative output 104.

Where the input transport stream 1 has a protocol which uses data packets which are longer than those of the data packets which are used by the second protocol and which are to be output via the alternative stream output interface, a different approach is adopted. As with the first situation, it is preferred that the transport controller core 124 reformat the transport packets but it is possible in embodiments of the present invention to have a dedicated processor for reformatting the transport packets. The received transport packet which is to be modified will provide, for example, one full packet and one incomplete packet. As with the previously described situation, it may be necessary to change the positioning and/or order of the received bytes or bits of the

received data packet having the first protocol in the reformatted packets having the second protocol.

The reformatted packet which is filled with the data is, as with the previous situation, passed to the multiplexer 134 along with a control signal via interconnect 136. A control signal is also passed to the select data block 140 via interconnect 172. The select data block 140 and further FIFO 138 will operate in a similar manner to that described in relation to the previous situation.

The next packet having the second protocol which contains the bits or bytes which were unable to fit into the previous packet, in one embodiment, are filled up with stuffing bits to fill the packet. The second packet is then output to the multiplexer 134 along-with the associated control signals via interconnect 172 and interconnect 136 to allow the second packet to be output via the output 104. The second packet is preferably output when the next synchronisation byte is received by the latency block 142. In other words, the first and second packets are preferably output on the receipt of consecutive synchronisation bytes by the packet start block 120.

However, in one modification to the invention, the transport controller core 124 stores the second, incomplete packet having the second protocol and waits until the next transport packet in the first protocol containing information relating to the programme to be output via output 104 is received. The second packet is then filled up with bits from this subsequent transport packet and then output to the multiplexer 134 with the associated control signals. The remainder of the second transport packet having the first protocol and including information relating to the selected programme is reformatted to be included in a data packet having the second protocol. If the reformatted packet is full, it will be output to the multiplexer 134 with the associated control signals to the multiplexer 34 and the select data block. On the other hand, if the packet is not full, then

transport controller core 124 will wait for the next packet having the first protocol and containing information relating to the selected programme and then fill that packet.

With this arrangement, the transport controller core may wait only a predetermined time for the next packet having the first protocol and containing information relating to the selected programme and if such a packet is not received in that time, stuffing bits may be added to the incomplete packet to fill it. The filled packet would then be output as described hereinbefore. Alternatively, if the required packet having the first protocol is not received within the predetermined time, then the incomplete packet may be discarded.

It should be appreciated that with some protocols, it is only necessary to alter the header of the data packets in order to provide the re-protocolled packets. With these protocols, the data itself, i.e. the payload is unaltered. When changing a packet having one protocol into a packet having a different protocol, it may be necessary, depending on the two protocols in question, to alter the values of specific bits or bytes of the header and/or payload so that the new packet conforms to the different protocol.

With embodiments of the present invention, it is preferred that the latency is such that the re-protocolling of the packets occurs in real time, although this is not necessary if only a part of the received transport stream is to be output via the alternative output stream 106.

It should be appreciated that it is also possible for the PTI 10 to output a programme with the received protocol via interconnect 502 for viewing by a user whilst a different or even the same programme is being output in a different protocol on the alternative output stream 106 and being recorded. It should be appreciated, that the alternative output having a different protocol may be required even if a recording device is not

connected to the alternative output 104. Other devices may alternatively be connected to the alternative output 104.

It is of course possible that the protocol of the received stream of data may use packets which are of the same length as the protocol to be used for the alternative output stream. In those circumstances, the transport controller core would simply change the format of each packet of data to provide a new packet in the required different protocol.

It should be noted that in preferred embodiments of the present invention, the protocol of the received transport stream 1 and the protocol of the alternative output stream both comply with the MPEG-2 standard.

Thus, the PTI 10 may be able to receive transport streams having a range of different protocols and standards and the PTI 10 will be able to decode data packets in the transport stream according to the correct protocol. The protocol used will generally be identified from the packet header. The PTI 10 may be able to re-protocol at least part of the received stream to a protocol with which a recorder or the like connected to the alternative output 104 is able to deal.

It may in certain circumstances be desirable to provide an encoded output from the alternative stream interface 104. In some embodiments of the present invention, the encoded output will have the same protocol as the received transport stream 1 whilst in other embodiments of the present invention, the encoded output from the alternative stream interface 104 will have a different protocol from that of the received transport stream 1.

For example, parents may wish to prevent children from viewing unsuitable material which has been recorded by the parents for viewing by themselves. Providers of, for example, a pay television service, may wish to avoid the problem of an unscrupulous user recording a programme and subsequently

duplicating for commercial purposes that programme. Providers of a pay television device may wish that a user be able to view a recorded programme only via his own set top box, thus preventing the user from lending recorded material to friends etc. Alternatively or additionally, it may be desirable that a recorded program may be viewed using the recording machines of third parties only if the user who has recorded the programme provides a descramble key such as, for example, a PIN (personal identification number) number.

The situation where the user wishes to prevent, for example, children from viewing unsuitable material will now be discussed. The user will, when he wishes to record a programme in scrambled form input an input PIN number or even just a simple indication that a programme is to be recorded in an encoded form via input 144. This input is passed to the transport controller core 124 via interconnect 174 which notes that the output provided via alternative stream output interface 104 is to be scrambled. As can be seen from Figure 3, there are a number of different locations where additional scramblers 146,148 or 150 can be provided in the input interface 100.

Where a PIN number has to be input by a user, that may be altered by the user each time he wishes to record a programme. Alternatively, the number may be unique to a particular PTI 10. In this latter case, when the user receives the set top box including the PTI 10, he will also be given the PIN number associated with that device. If the PIN number is unique to the set top box, then the user would only need to provide an indication (and not the PIN number) that a scrambled alternative output is required.

In one embodiment, a scrambler 150 is provided between the output of the multiplexer 134 and the further FIFO 138. Scramblers 146 and 148 may be omitted. In those circumstances, the output of the FIFO 122, is directly connected to the second multiplexer 134. The output of the descrambler 132 is also connected to the second

multiplexer 134. A control signal is provided by the transport controller core 124 via interconnect 136. If appropriate, a reformatted packet input 152 may also be provided to the second multiplexer 134. The control signal provided by the transport controller core 124 is arranged to select one of the three inputs to the second multiplexer 134. The input of the multiplexer 134 which connects the output of the FIFO 122 directly to the multiplexer 134 is selected if the received transport packet to be output via the alternative stream output interface does not contain any scrambled data and no change to the protocol is required. The input of the multiplexer 134 which connects the output of the descrambler 132 to the multiplexer 134 will be selected if the transport packet to be output via the alternative stream output interface 104 initially contains scrambled data and no change to the protocol is required. Finally, the reformatted data packet input 152 will be selected if a reformatted output is required.

The output of the second multiplexer 134 is connected to the scrambler 150, which is controlled by the transport controller core 124 via interconnect 174. The output of the second multiplexer 134 is then scrambled by the scrambler 150 in accordance with a scramble key supplied by the transport controller core 124. The scrambled data is then output to the FIFO 138. The FIFO 138, the select data block 140 and the latency block 142 operate in the same manner described in relation to the reprotocolling of the transport packets. In particular, it is preferable that the select data block 140 allow the scrambled transport packet to be output when the synchronisation byte for the subsequent packet is received. The latency is also controlled in a similar manner to that described in relation to the reprotocolling of the transport packets.

In an alternative embodiment of the invention, scrambler 150 is omitted and instead, scramblers 146,148 are used. The first scrambler 146 is provided in the path between the FIFO 122 and the multiplexer 134. In other words, the output of the FIFO 122

is connected directly to the input of the first scrambler 146. The second scrambler 148 is connected between the output of the descrambler 132 and the input of the multiplexer 134. The same criteria for selecting the various inputs to the multiplexer 134 also applies in this embodiment. However, the packet which is applied to the multiplexer 134 will be scrambled by virtue of the first scrambler 146 or the second scrambler 148. In this modification, the reformatted packet, if selected, would not be scrambled. Again, the first and second scramblers 146 and 148 are controlled by the transport controller core 124 which is arranged to control how the received packet is scrambled. Again, the FIFO 138, the select data block 140 and the latency block 142 operate in the same manner as described in relation to the previous embodiment.

The data packets can be scrambled using any suitable method. However, it is generally preferred that a relatively simple method be used. In other words it is preferred that the encryption technique used does not use a strong encryption method. For example, the PID number could be scrambled, toggled or inversed so that the receiver is unable to correctly identify the received data packet unless it has the descramble key. It is also possible to flip certain key bits of the transport packet, for example in the transport packet header, which again prevent a receiver from being able correctly to interpret the transport packet. The header bytes may be shifted. Alternatively the order of some of the header bytes may be altered. One example of scrambling is transport scrambling. Transport scrambling can take place in any stream. The status of a scramble flag is signalled along with the key (even or odd) with which it has been scrambled via two bits in the transport header. The payload is then scrambled.

It is also possible to re-encrypt part of PES data (Packetised Elementary Stream) which forms one of the protocols of the MPEG standard. A PES packet is defined by the MPEG 2 standard and the PES header and payload are included in a packet which also

include a start part, a stream identification part and a PES packet length part.

PES scrambling is performed on PES data which includes PES headers and PES payloads. An equivalent 2 bit field in the PES header is set to indicate that the packet is scrambled and the key which is used. The PES payload is scrambled. A first transport packet in a different protocol would then contain the PES header and some of the PES data. The next transport packet(s) would contain any remaining of the PES data.

Another possibility for scrambling the packet is to scramble for example the adaption field control. This is one of the elements which is sometimes provided in packet headers.

Part or all of the descramble key may be carried in the transport packet output via the alternative stream output interface to allow the recorded material to be descrambled by the recording device when replaying the recorded material.

In this regard the descramble key may be made up of one or more of the following: identification from the set top box; a smart card, if provided; and/or the PIN number entered by a user.

Where the user wishes to prevent, for example a child from viewing unsuitable recorded material, the user would input via input 144 either a PIN number or a simple indication that the material is to be recorded in a scrambled form. If a PIN number is associated with the set top box, then there is no need for the user to input that number in order to achieve scrambling. The transport controller core 124 will control the respective scrambler(s) in order to ensure that the selected programme is correctly scrambled and that the transport packet if required includes a descramble key. This may include the PIN number. The packets of data in relation to the programme to be scrambled are identified by the transport controller core 124 in the same manner as described in relation to the change in protocol. The

transport packets are scrambled using any of the methods outlined hereinbefore or any other suitable method by the respective scrambler under the control of the transport controller core 124.

Where the user has recorded a programme which is not to be viewed by unauthorised people such as a child, the data stream relating to one programme will not be unscrambled unless the user inputs the correct PIN number. The correct PIN number would need to be input into the recorder in order to descramble the recorded packet. Alternatively, the recorded programme will, when played by the recorder, provide an input stream to the programmable transport interface 10 itself. The PIN number would then be input via input 144. The transport control core 124 may itself unscramble the packets scrambled by scrambler 146, 148 and/or 150. Alternatively the descrambling may be carried out by the descrambler 132. In either case, the PIN number would be required in order to descramble the recorded packets. This would, for example, prevent children from viewing unsuitable material as the video would only be watchable if the correct PIN number was input to the programmable transport interface 10 or the recorder.

In other embodiments of the present invention, the scrambler would automatically scramble any output provided on the alternative stream output interface 104 in accordance with instructions from the transport controller core 124. The recorded material would only be viewable if the recorded material were to be played back so that it provided the transport stream input into the input interface 100 of the same set top box which was initially used to record the material. The transport controller core would thus be able to descramble the recorded material as, it is able to provide the necessary descramble keys. This would prevent the user from recording material and giving it to third parties. With this arrangement, no PIN number or input 144 would be required.

In yet another modification, any material recorded would be scrambled. The recorded material would be viewable through the

user's own recorder and/or set top box and also viewable using any other suitable recorder and/or set top box provided that the user inputs the correct PIN number. In the first situation, if the recorded material is replayed back through the user's own set top box, the set top box may be arranged to identify that the recorded material has been scrambled by itself. In those circumstances, the set top box may not require the user to enter the PIN number but will automatically descramble the recorded packets. In the latter circumstances, the user will need to input the PIN number before the recorded packets can be descrambled either by the recorder or another set top box. Information on the PIN number would be included in the scrambled output packets.

It should be appreciated that in embodiments of the present invention, the reformatting of the protocols of the data packet can be omitted. In alternative embodiments of the present invention, scramblers 146,148,150 may be omitted.

The signals provided via interconnects 160, 164, 166, 168, 136 and 172 correspond to the control signals provided by interconnect 112 of Figure 2.

It should also be appreciated that it is possible that the PTI 10 will output a programme to be viewed by a user via interconnect 502 at the same time that a scrambled output is output via the alternative stream output 104. That scrambled output may contain the same or a different programme to that being viewed by the user in real time.

Reference will now be made to figure 4 which schematically shows set top box 701 which includes a programmable transport interface 10, as shown in figure 2. The output of the programmable transport interface 10 is connected to the MPEG-2 decoder 702. The MPEG-2 decoder 702 forms part of the set top box 701. The output of the MPEG-2 decoder provides an output of the set top box 701 and is connected, for example, to a display 704.

The alternative output 104 of the PTI 10 is connected to a recorder 706 which may record a scrambled and/or reprotocolled data stream. For completeness sake, the channel decoder 708 of the set top box 701 is also shown. The output of the channel decoder 708 provides the input to the input interface 102 of the programmable transport interface 10. The MPEG-2 decoder 702 and the programmable transport interface 10 together define the source decoder. As mention hereinbefore, the system processor 700 is able to vary the instructions for the transport controller of the programmable transport interface.

In preferred embodiments of the present invention, the transport controller core is programmable so that the transport controller core can be programmed to change the protocols with which the transport controller core can deal. The reprogramming of the transport controller core 124 is achieved by changing the instructions and data in the SRAMs 400 and 300. However, in alternative embodiments of the invention, the transport controller instructions and data could be hard coded. In other words hardware could be used.

It should be noted that in some embodiments of the present invention, the received transport stream can be output via the alternative stream output 104. That transport stream may be processed, for example, to be descrambled where necessary or unprocessed.

With both of a scrambled and a reprotocolled output provided on the alternative output stream, it is preferred that real time processing of the input stream occur to provide the alternative output stream. It is also preferred that there be a fixed latency between the input 102 of the input interface 100 and the alternative output 104 of the input interface 100.

In one modification to the embodiment described hereinbefore, a number of different descramblers are provided in the input interface 100, in parallel. One descrambler will be provided for

each different standard or protocol which is capable of being received by the PTI 10. The input stream will be directed, for example by means of a multiplexer, under the control of the transport controller core to the descrambler which is able to deal with particular protocol or standard of the received transport stream. The transport controller core 124 has been described as not performing any descrambling itself. In some embodiments, the transport controller core 124 may carry out descrambling, for example to remove weak encryption of the data.

In the present case, the term programme is used to cover, television programmes, subtitles, teletext, films, audio programmes, files of data, audio and/or visual information such as collections of music, etc or the like.

In the embodiment described herein, the alternative stream output is described as being connected to a recorder. However, the alternative stream output can be corrected to any other suitable device such as a screen, a digital video recorder, a PC, another set top box, a network connector or the like. For example, the alternative stream output may be connected to an IEEE-1394 interface so that any device compatible with the IEFE 1394 interface can be connected thereto. Of course other types of parallel ports can be used instead of the IEEE 1394 interface provided that the port is capable of sustaining the data rate needed by a transport stream.

The transport controller 104 is preferably a microprocessor or the like. The whole of the programmable transport interface may be incorporated in an integrated circuit.

In embodiments of the present invention where the packets of the first protocol are shorter than the packets of the second protocol, packets of the first protocol may be split across two packets of the second protocol in a similar manner to the case described where the packets of the first protocol are longer than those of the second protocol.

In one embodiment of the invention, the alternative output of 106 of the PTI 10 may be in the ATM format (Asynchronous Transmission Mode). The entire transport packet including the header and the payload will become the payload of an ATM stream. A wrapper would then be added to data which can then be sent to an ATM network.

It should be appreciated that embodiments of the invention can be used in applications other than set top boxes. For example the PTI may be included in an ATM receiver or the like.

It should be noted that the term protocol should be broadly interpreted. In the protocol conversion aspect of the hereinbefore described embodiment, the first and second protocols may be with the same standard. For example the conversion may be between standards eg from the transport stream protocol to the program stream protocol in the MPEG-2 Standard. Alternatively the first and second protocols may be in different standards. For example the MPEG-2 Transport stream could be converted to an ATM stream. It is thus clear that the first and second protocols do not necessarily conform to the same standard.

Latency is used in the PTI to maintain a fixed delay between the data coming into the PTI and the data leaving via the alternative output. Latency is generally only of importance if the reprotocolling requires the relative timing for the data being received and output via the alternative output to be maintained such as with DVB outputs.

CLAIMS:

1. A device for receiving a stream of data having a first protocol, said device comprising:

means for selecting at least a portion of data to be output in a second, different protocol;

processing means for processing the selected data so that said selected data has said second protocol; and

means for outputting the selected data having the second protocol.

2. A device as claimed in claim 1, wherein whilst said means for outputting said selected data outputs said data, the processing means is arranged to process the next selected portion of data.

3. A device as claimed in claim 1 or 2, wherein said stream of data comprises a stream of packets of data.

4. A device as claimed in claim 3, wherein the length of the data packets of the first protocol are shorter than the length of the data packets of the second protocol, and the processing means is arranged to include data from a packet of said selected data having said first protocol and stuffing bits in a packet of data having said second protocol, said output means being arranged to output said packet of data in the second protocol.

5. A device as claimed in claim 3, wherein the length of the data packets of the first protocol are longer than the length of the data packets of the second protocol, and the processing means is arranged to include data from a packet of said selected data having the first protocol in a plurality of packets of data having said second protocol, said output means being arranged to output said packets of data in the second protocol.

6. A device as claimed in claim 5, wherein one of said

plurality of packets in said second protocol is not full with data from said packet of selected data in said first protocol and said processor means is arranged to include stuffing bits in said packet which is not full to thereby fill that packet.

7. A device as claimed in claim 5, wherein one of said plurality of packets in said second protocol is not full with data from said packet of selected data in said first protocol, and said processor means is arranged to include data from the next packet of said selected data to thereby fill said one packet.

8. A device as claimed in any preceding claim, wherein said data stream includes a plurality of different programmes.

9. A device as claimed in claim 8, wherein said programmes are television programmes.

10. A device as claimed in claim 8 or 9, wherein only one of said plurality of programmes is output by said outputting means.

11. A device as claimed in claim 9 or 10, wherein a selected program is output on a different output of said device.

12. A device as claimed in claim 11, wherein the programme output from said output means is output at the same time as the programme output from said different output.

13. A device as claimed in claim 3 or any claim appended thereto, wherein the packets of the first protocol and the second protocol each have a header part and a payload part and when converting the packets of the first protocol to packets of the second protocol, the header part of the packet having the first protocol is modified to conform to the second protocol.

14. A device as claimed in claim 3 or any claim appended thereto, wherein detecting means are provided for detecting the

beginning of each packet having said first protocol and the timing of said device is dependent on the detection of the beginning of each packet.

15. A device as claimed in claims 13 and 14 wherein said detecting means are arranged to detect at least a portion of the header to detect the beginning of the packet.

16. A device as claimed in any of the preceding claims, wherein said selected data is processed in real time.

17. A device as claimed in claim 16, wherein a packet of data having the first protocol is converted by the processing means to at least one packet having the second protocol before the next packet of data having the first protocol is received by said receiving device.

18. A device as claimed in claim 17, wherein a packet of data in the second protocol is output when the beginning of the next packet in the first protocol is received.

19. A device as claimed in any preceding claim, comprising an input interface having an input for receiving said stream of data and said outputting means and a controller having said processing means.

20. A device as claimed in claim 19, wherein said input interface comprises means for descrambling the received stream of data.

21. A device as claimed in claim 19 or 20, wherein said input interface is controlled by said controller.

22. A device as claimed in claim 19, 20 or 21, wherein said detecting means is arranged to provide an output signal to the processor to control the timing of the device.

23. A device as claimed in any preceding claim, wherein said output means include buffer means.

24. A device as claimed in any preceding claim, wherein said processing means comprises a programmable processor.

25. A device as claimed in any preceding claim wherein said data stream comprises a digital data stream.

26. A device as claimed in claim 25, wherein said first and second protocols both conform to the MPEG-2 standard.

27. A device as claimed in any preceding claim, wherein said device forms part of a set top box.

28. A device for use in a set top box comprising:

means for receiving a stream of packets of data having a first protocol and including information relating to a plurality of different programmes;

means for processing the received stream of data to identify selected packets of data containing data relating to a programme, said processing means being arranged to process the selected data packets so the selected packets conform to a second, different protocol;

output means for outputting the selected data packets in the second protocol.